#12 Appeal Brief 2-6-03

IN THE UNITED STATES PATENT OFFICE

| | | | |
|---|---|---|---|
| Applicants: | Reto Stamm et al. | | |
| Assignee: | Xilinx, Inc. | | |
| Title: | SYSTEM AND METHOD FOR TESTING PARAMETERIZED LOGIC CORES | | |
| Ser. No.: | 09/360,472 | Filing Date: | 07/23/1999 |
| Examiner: | Thai Q. Phan | Art Unit: | 2123 |
| Docket No.: | X-528-US | Conf. No.: | 4229 |



COMMISSIONER FOR PATENTS
Washington, D.C. 20231

Sir:

This is an Appeal Brief submitted pursuant to 37 C.F.R. §1.192 for the above-referenced patent application and is being filed in triplicate.

## I. Real Party in Interest

The real party in interest is Xilinx, Inc., having a place of business at 2100 Logic Drive, San Jose, California 95124-3400. The above referenced patent application is assigned to Xilinx, Inc.

## II. Related Appeals and Interferences

There are no related appeals or interferences.

## III. Status of Claims

Claims 1-20 are presented for appeal. Claims 1-20 stand rejected under §102(e) as being anticipated by *Dockser et al.* (U.S. Patent No. 5,963,454). The claims presented for appeal, as presently amended, may be found in the attached Appendix of Appealed Claims.

## IV. Status of Am ndments

The application was initially filed on July 23, 1999, including claims 1-20. In reply to a first Office Action, which was mailed on April 11, 2002, a Response was filed on June 21, 2002 and no claims were amended. A final Office Action was mailed on October 3, 2002. An interview by telephone was conducted on November 27, 2002, and no agreement was reached. A Notice of Appeal was filed via first class mail on December 3, 2002, and an Advisory Action was mailed on December 23, 2002.

## V. Summary of Invention

One embodiment of Appellant's invention is directed to a method for testing a parameterizable logic core. A set of parameter values for the logic core is randomly generated, and a netlist is generated from the parameter values. Circuit behavior is then simulated to test the logic core (page 2, ll. 15-20).

In another embodiment, random parameter values are provided as input to a graphical user interface (GUI) of a core generator. The capability of the GUI to respond to erroneous parameter sets is thereby tested (page 2, ll. 21-24).

In yet another embodiment, if simulation of circuit behavior in association with a particular parameter set results in an error, the parameter set is cloned and mutated to generate a netlist for another simulation. Cloning and mutating the parameter sets may be used to assist in replicating error conditions (page 2, ll. 25-30).

## VI. Issue for Review

**Is the rejection of claims in groups I-XII under 35 USC §102(e) over Dockser et al. (USP #5,963,454, hereinafter, "Docks r") proper when Dockser fails to teach or sugg st every limitation of the claim d invention?**

## VII. Grouping of Claims

For purposes of this appeal, claims 1, 18, 19, and 20 are in group I, claim 2 is in group II, claim 3 is in group III, claim 4 is in group IV, claims 5 and 15 are in group V, claim 6 is in group VI, claims 7 and 16 are in group VII, claims 8 and 10 are in group VIII, claims 9 and 10 are in group IX, claims 12 and 13 are in group X, claim 14 is in group XI, and claim 17 is in group 12. The claims as now presented in the different groups do not stand or fall together.

## VIII. Argument

**The §102(e) rejection of claims in groups I-XII is not proper when Dockser fails to teach or suggest every limitation of the claimed invention**

In order to establish a *prima facie* case of anticipation, the Examiner must present a reference that completely corresponds to the claimed invention.

The claims of group I include limitations that relate to randomly generating a set of parameter values for a logic core, generating a netlist from the logic core and randomly generated parameter values, and simulating a circuit using the netlist. Dockser teaches a tool for developing an integrated circuit design (Abstract). Dockser's tool creates an HDL template, creates a parameter file, encrypts the HDL template, …, creates HDL for synthesis, and creates netlists for the HDL (Abstract). Dockser clearly neither teaches nor suggests the claim limitations of randomly generating a set of parameter values. Specifically, Dockser teaches:

> In a step 304, a parameter file and a parameter check file are created for each HDL template. The portion of each HDL template which pertains to parameters is obtained from the HDL template and used to create a parameter file. As such, the parameter file includes information used to define global parameters for the HDL template with which the parameter file is associated. As will be described

> below with respect to FIG. 4, the parameter file typically also includes information which may be used to define a user input window for the HDL template. That is, the parameter file is typically used to define a Graphical User Interface (GUI). **(col. 6, ll. 25-36).**
> ...
> After the menu is created and displayed, input is received from the user in a step 404. The input may include, but is not limited to, design specifications in the form of parameters, requests to perform operations using parameters, and requests to abort the process of developing design specifications. **(col. 6, ll. 59-64).**

These teachings clearly show that Dockser neither teaches nor suggests the claim limitations. Instead of randomly generating parameter values as claimed, a user of Dockser's system enters design specifications, presumably directed to a particular implementation of a circuit design. Therefore, the claims of group I are not anticipated by Dockser, and the rejection should be withdrawn.

Claim 2 in group II includes limitations that relate to generating a random parameter value within predetermined upper and lower limits, and generating a new random parameter value if the random parameter value fails to meet predetermined criteria. The cited section of Dockser that is alleged to teach these limitations discusses checking whether input parameters are out of range (col. 7, ll. 9-24). This is not identical to the limitations of generating a random parameter value within a range. Therefore, the claim of group II is not anticipated by Dockser, and the rejection should be withdrawn.

Claim 3 of group III includes limitations that relate to assigning respective probabilities to numbers between upper and lower limits for the parameters and generating the random parameter value as a function of the probabilities. The rejection alleges that these limitations are taught in Dockser's col. 8, l. 56 - col. 9, l. 32. However, this section of Dockser teaches that "the parameter file is traversed to find all possible permutations of parameter combinations, which are then used to create a list of desired options sets."

Dockser does not appear in any way to teach or suggest assigning probabilities as claimed.

The rejection further alleges that "Dockser discloses parameter values are generated and to account for design change or modification it would be necessarily require the randomness of parameter values … [which] would take a distribution function or probability function as claimed such that the generation of parameter values would be typically feasible and meet design environment." As far as this explanation is understood, the reasoning appears to be contrary to generation of random parameter values. If a designer is trying to create a suitable design, the designer would not generate random parameter values. Rather the designer would use his or her experience to select suitable parameter values. Therefore, Dockser neither teaches nor suggests assigning probabilities and the related limitations. Claim 3 of group III is not anticipated by Dockser, and the rejection should be withdrawn.

Claim 4 of group IV includes limitations that relate to providing a parameter value as input to a GUI and generating random replacement values for invalid values detected by the GUI. The rejection alleges that Dockser's col. 2, ll. 44-59 and col. 12, ll. 1-65 teach these limitations. Dockser's col. 2 ll. 44-59 discusses the use of macro templates and RTL to enable a user to modify parameters in the macro template prior to synthesis. There is no mention of random generation of parameter values in this section. Dockser's col. 12 includes Appendix I and Appendix II which set forth code for a "fifo template compiler" (Appendix I) and code that appears to check parameter values (Appendix 2), respectively. Again there is no apparent teaching of generating random parameter values and generating random replacement values for invalid values detected by the GUI. Instead, Dockser appears to teach against random generation of replacement parameter values in view of Dockser's output of error messages when a value is out of range

(col. 12, ll. 65-66). Since Dockser outputs an error message, it would appear to be for the purpose of guiding a user toward suitable parameter values, not a random generation of parameter values. Therefore, the claim of group IV is not anticipated by Dockser, and the rejection should be withdrawn.

The claims of groups V, VI, VII, VIII, IX, X, XI, and XII were generally rejected on the basis of Dockser's col. 8, l. 56 - col. 9, l. 32. These claims include various limitations that are clearly neither taught nor suggested by Dockser. For example, claims 5 and 15 in group V include limitations that relate to providing the randomly generated set of parameter values to a GUI and identifying invalid parameter values with the GUI. The cited portions of Dockser do not appear to teach providing randomly generated parameter values to a GUI. Therefore, the claims of group V are not anticipated by Dockser, and the rejection should be withdrawn.

Claim 6 of group VI includes limitations that relate to generating random replacement parameter values for invalid parameters and repeating the steps of providing the parameter values to the GUI, identifying invalid parameter values, and generating random replacements until all the parameter values are valid. As explained above in regards to the claims of group IV, Dockser's output of error messages would appear to teach against random generation of parameter values as claimed. Therefore, the claim of group VI is not anticipated by Dockser, and the rejection should be withdrawn.

Claims 7 and 16 of group VII include limitations that relate to selecting a random order in which to provide parameter values to a GUI, and providing the values one-by-one as input. The rejection is improper because no correspondences between elements of Dockser and these limitations have been provided. Furthermore, Dockser does not appear to select a random order for input of parameter values. Therefore, the

claims of group VII are not anticipated by Dockser, and the rejection should be withdrawn.

The claims of group VIII, IX, X, and XI include limitations that relate to cloning and mutating sets of parameter values. No portions of Dockser are alleged nor does Dockser appear to teach or suggest these limitations. Therefore, the claims of group VIII and IX are not anticipated by Dockser, and the rejection should be withdrawn.

The claim of group XII includes limitations that relate to highlighting parameters where a particular parameter value causes every test that is run with that parameter value to fail. No portions of Dockser are alleged nor does Dockser appear to teach or suggest these limitations. Therefore, the claim of group XII is not anticipated by Dockser, and the rejection should be withdrawn.

Without complete correspondence, the §102 rejection cannot stand. Accordingly, Appellant submits that the §102 rejection is improper and the rejection must be withdrawn.

Claims 1, 18, 19, and 20 of group I are separately patentable over the claims in the other groups because the limitations of the claims in the other groups are not necessarily present in the group I claims and the limitations of the group I claims are not taught by the prior art.

Claim 2 of group II is separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than group I) are not necessarily present in claim 2, and the limitations of generating a random parameter value within predetermined upper and lower limits and generating a new random parameter value if the random parameter value fails to meet predetermined criteria are not necessarily present in the claims of the other groups (other than groups III and IV), nor are the limitations taught by the prior art.

Claim 3 of group III is separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than groups I and II) are not necessarily present in claim 3, and the limitations of assigning respective probabilities to numbers between upper and lower limits for the parameters and generating the random parameter value as a function of the probabilities are not necessarily present in the claims of the other groups (other than group IV), nor are the limitations taught by the prior art.

Claim 4 of group IV is separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than groups I, II, and III) are not necessarily present in claim 4, and the limitations of providing a parameter value as input to a GUI and generating random replacement values for invalid values detected by the GUI are not necessarily present in the claims of the other groups, nor are the limitations taught by the prior art.

Claims 5 and 15 of group IV are separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than group I) are not necessarily present in claims 5 and 15, and the limitations of providing the randomly generated set of parameter values to a GUI and identifying invalid parameter values with the GUI are not necessarily present in the claims of the other groups (other than in dependent claims), nor are the limitations taught by the prior art.

Claim 6 of group VI is separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than groups I and V) are not necessarily present in claim 6, and the limitations of generating random replacement parameter values for invalid parameters and repeating the steps of providing the parameter values to the GUI, identifying invalid parameter values, and

generating random replacements until all the parameter values are valid are not necessarily present in the claims of the other groups (other than dependent claim 7 of group VII), nor are the limitations taught by the prior art.

Claims 7 and 16 of group VII are separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than parent claims) are not necessarily present in claims 7 and 16, and the limitations of selecting a random order in which to provide parameter values to a GUI and providing the values one-by-one as input are not necessarily present in the claims of the other groups, nor are the limitations taught by the prior art.

Claims 8 and 10 of group VIII are separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than parent claims) are not necessarily present in claims 8 and 10, and the limitations of cloning and mutating sets of parameter values are not necessarily present in the claims of the other groups (other than dependent claims), nor are the limitations taught by the prior art.

Claims 9 and 11 of group IX are separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than parent claims) are not necessarily present in claims 9 and 11, and the limitations of conditioning the cloning and mutating of parameter value sets on whether an error is detected are not necessarily present in the claims of the other groups (other than dependent claims), nor are the limitations taught by the prior art.

Claims 12 and 13 of group X are separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than parent claims) are not necessarily present in claims 12 and 13, and the limitations of repetitively cloning and mutating sets of parameter values and mutating one or more of the parameter values in each set

are not necessarily present in the claims of the other groups (other than in the dependent claims), nor are the limitations taught by the prior art.

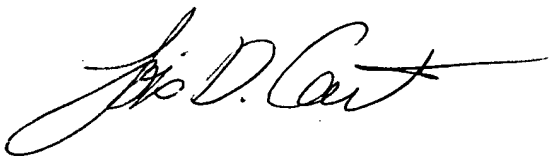
Claim 14 of group XI is separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than parent claims) are not necessarily present in claim 14, and the limitations of the number of parameter values mutated being a function of the number of previous generations are not necessarily present in the claims of the other groups, nor are the limitations taught by the prior art.

Claim 17 of group XII is separately patentable over the claims in the other groups because the limitations of the claims in the other groups (other than parent claims 1) are not necessarily present in claim 17, and the limitations of highlighting parameters where a particular parameter value causes every test that is run with that parameter value to fail are not necessarily present in the claims of the other groups, nor are the limitations taught by the prior art.

## IX. Conclusion

In view of the above, Appellant believes the claimed invention to be patentable. Claims 1-20 remain for consideration. Appellant respectfully requests reversal of the rejections as applied to the appealed claims and allowance of the entire application.

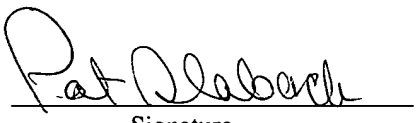
Respectfully submitted,



Lois D. Cartier
Agent for Applicants
Reg. No.: 40,941

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, Washington, D.C. 20231 on January 30, 2003.

Pat Slaback
Name



Signature